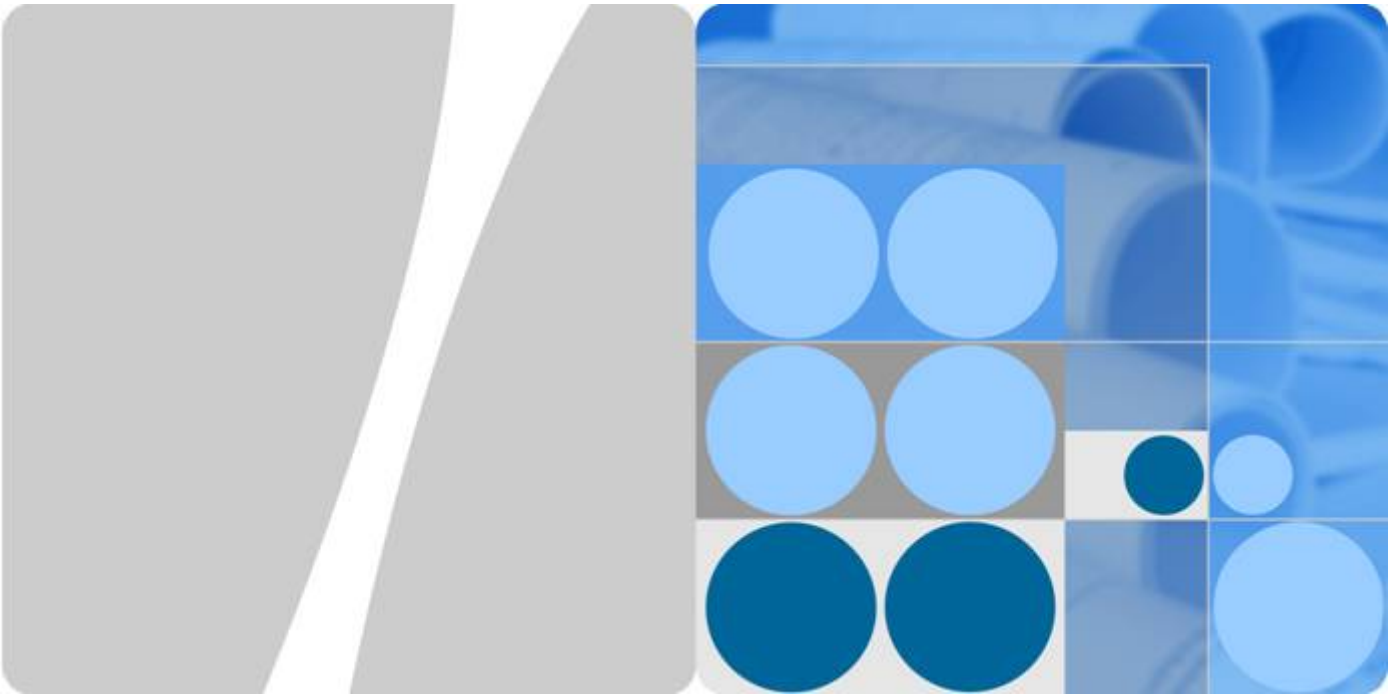


Object Storage Service

Tool Guide (obsfs)

Date 2021-06-30

Contents

1 Introduction to obsfs 1
2 Environment Preparation 3
2.1 Resource Preparation 3
2.2 Downloading and Installing obsfs 6
2.3 Initializing obsfs 7
3 Operation Guide 9
3.1 Mounting a Parallel File System 9
3.2 Unmounting a Parallel File System 14
4 FAQs 16
5 Troubleshooting 18
5.1 Failure in Mounting a Parallel File System 18
5.2 Failure in Unmounting a Parallel File System 20
6 Related Operations 21
6.1 libfuse Installation Guide 21
A Change History 23

1 Introduction to obsfs

obsfs, built on the basis of Filesystem in Userspace (FUSE), is a file system tool provided by Object Storage Service (OBS) for mounting OBS parallel file systems to Linux operating systems. It enables you to easily access the infinite storage space on OBS in the same way as you operate the local file system.

If you are used to storing data in local file systems and do not want to change the way of data accessing even though data is stored in OBS, obsfs is recommended.

Functions

- Supports the mounting of parallel file systems to your local file system on the Linux OS, so that you can use your local file system to manage objects stored on OBS.
- Supports synchronous upload. Any file uploaded to the mounted directory can be synchronously uploaded to OBS.
- Supports synchronizing objects from a parallel file system to its mounted directory in the local file system, so that you can copy, modify, rename, and truncate objects locally.

Constraints

- The local directory to which the parallel file system is mounted needs not to be empty. But its original files will be unavailable after the parallel file system is mounted. They are recovered when the parallel file system is unmounted. Therefore, it is recommended that you use an empty directory as the target directory for mounting a parallel file system.
- obsfs supports mounting of parallel file systems but not OBS buckets.
- The parallel file system mounted to the local system cannot provide the same performance and functions as the local file system.
  - Files or folders in the mounted directory do not support hard link commands.
  - The number of directory levels cannot exceed 45.

Applicable Operating Systems

obsfs is applicable to Linux operating systems. For details, see **Table 1-1**.

Table 1-1 Obtaining obsfs

Method	Description	Applicable Linux OS	Operation Guide
Download	Download the official software package, configure the runtime environment in the Linux OS, and then obsfs can be used.	Ubuntu 16 and CentOS 7	**Download and Install obsfs**

2 Environment Preparation

2.1 Resource Preparation

Before using obsfs, you need to prepare the resources listed in **Table 2-1**.

Table 2-1 Resource preparation

Resource	Description	Operation Guide
IAM user	To ensure account and resource security, you are advised to create an IAM user and assign permissions to the user for accessing OBS resources. Then use the IAM user to perform operations on obsfs.	**Creating an IAM User**
Access keys (AK and SK)	Access keys (AK and SK) are identity credentials of your account or IAM users. obsfs accesses OBS using access keys. Access keys are used to encrypt the signature of a request, ensuring that the request is secure and integral, and that identities of the request sender and receiver are correct.	**Creating Access Keys (AK and SK)**
Parallel file system	A parallel file system is a container for storing data.	**Creating a Parallel File System**

Creating an IAM User

To ensure account and resource security, you are advised to create an IAM user and assign permissions to the user for accessing OBS resources. Then use the IAM user to perform operations on obsfs.

Step 1 Log in to the management console using a cloud service account.

Step 2 On the top navigation menu, choose **Service List** > **Management & Deployment** > **Identity and Access Management**. The IAM console page is displayed.

Step 3 Create a user group and grant the OBS permissions to the user group.

User groups facilitate centralized user management and streamlined permission management. Users in the same user group have the same permissions. Users created in IAM inherit permissions from the groups to which they belong.

1. In the navigation pane on the left, click **User Groups**. The **User Groups** page is displayed.
2. Click **Create User Group**.
3. On the **Create User Group** page, enter a name for the user group and click **OK**.

   The user group is then displayed in the user group list.
4. Click **Assign Permissions** in the **Operation** column of the row where the created user group resides.
5. In the **Group Permissions** area, locate the row that displays **Global service** > **OBS**, click **Attach Policy** in the **Operation** column, select the policy name, and click **OK**.

   NOTE

   In the **Policy Information** area, you can view the details about the policy.

   Due to data caching, an RBAC policy and fine-grained policy involving OBS actions will take effect 10 to 15 minutes after it is attached to a user, a user group, or an enterprise project.

Step 4 Create a user.

1. In the navigation pane on the left, click **Users**. The **Users** page is displayed.
2. Click **Create User**.
3. Set the user information and click **Next**.

Table 2-2 User parameters

Parameter	Description
Username	The user name for logging in to the cloud service.
Credential Type	A credential refers to the identity credential used for user system authentication. In this example, password is selected. – **Password**: Used for accessing cloud services using the console or development tools. – **Access key**: Used for logging in to the cloud service using development tools. This credential type is more secure, and is recommended if the user does not need to use the console.
User Groups	The user will inherit the permissions granted to the user group. The default user group is **admin**, which has the administrator permissions and all of the permissions required to use all cloud resources.

Parameter	Description
Description	(Optional) brief description of the user

4. Select a type for password generation, set the email address and mobile number, and click **OK**.

Step 5 Use the created IAM user to log in to OBS Console and verify the user permissions.

----End

Creating Access Keys (AK and SK)

Access keys (AK and SK) are identity credentials of your account or IAM users. obsfs accesses OBS using access keys. Access keys are used to encrypt the signature of a request, ensuring that the request is secure and integral, and that identities of the request sender and receiver are correct. If you have access keys (AK and SK), skip this part.

NOTE

Each account or IAM user can create a maximum of two valid access keys.

Step 1 Log in to OBS Console.

Step 2 In the upper right corner of the page, select **My Credentials** under the username.

Step 3 On the **My Credentials** page, select **Access Keys** in the navigation pane on the left.

Step 4 On the **Access Keys** page, click **Create Access Key**.

NOTE

A user can create a maximum of two valid access keys.

Step 5 In the **Create Access Key** dialog box that is displayed, enter the password and its verification code.

NOTE

- If you have not bound an email address or mobile number, enter only the password.
- If you have bound an email address and a mobile number, you can verify through either one.

Step 6 Click **OK**.

Step 7 In the **Download Access Key** dialog box that is displayed, click **OK** to save the access key to your browser's default download path.

NOTE

To prevent the access keys from being leaked, keep them secure. If you click **Cancel** in the dialog box, the access keys will not be downloaded, and you cannot download them later.

Step 8 Open the downloaded **credentials.csv** file to obtain the access keys (AK and SK).

NOTE

In the access key file, the value in the **Access Key ID** column is the AK, and the value in the **Secret Access Key** column is the SK.

----End

Creating a Parallel File System

For details about how to create a parallel file system, see **Creating a Parallel File System**.

2.2 Downloading and Installing obsfs

For common Linux distributions, such as CentOS 7 and Ubuntu 16, obsfs software packages are available for users to download and use. After downloading the obsfs tool, configure the operating environment and then run the mount command on the Linux operating system to mount the parallel file system.

Downloading obsfs Software Packages

Table 2-3 obsfs download links

Linux Distribution	Download Link
CentOS 7	**https://obs-community-obsfs.obs.ru-moscow-1.hc.sbercloud.ru/obsfs/current/obsfs_CentOS7.6_amd64.tar.gz**
Ubuntu 16	**https://obs-community-obsfs.obs.ru-moscow-1.hc.sbercloud.ru/obsfs/current/obsfs_Ubuntu16.04_amd64.tar.gz**

Installing obsfs

1. On a PC running Windows, download the obsfs software package based on the version of the Linux distribution.
2. Use a common cross-platform transmission tool (such as WinSCP) to transfer the software package to your Linux OS.
3. Run the following decompression command on the computer running Linux:
   tar -xzvf *absolute path of the obsfs software package* **/obsfs_***xxx***.tar.gz**

   NOTE

   You can also run the following command to decompress the software package to a specified directory:

   tar -xzvf *Absolute path of obsfs software package/obsfs_xxx.tar.gz* **-C** *Specified directory*
4. Go to the directory where the software package is decompressed and run the following command to install obsfs:
   ```
   bash install_obsfs.sh
   ```

> **NOTE**
>
> After you run the installation command, crontab will add a scheduled task for compressing and dumping obsfs logs. The dump path is **/var/log/obsfs/**.
>
> An example of such task added by crontab: ***/10 * * * * [-f /opt/dfv/obsfs/obsfs_log_rotate.sh] && bash /opt/dfv/obsfs/obsfs_log_rotate.sh**

Configuring Operating Environment

The operating environment of obsfs depends on the software packages such as openssl-devel, fuse, and fuse-devel. Before running obsfs, you need to run commands to configure the dependent environment.

> **NOTE**
>
> - obsfs supports libfuse 2.9.7, 2.9.8, or 2.9.9. If your libfuse is of another version, install it by referring to **libfuse Installation Guide**.
> - You can view the libfuse version by running the version query command on the CLI of the operating system. Alternatively, you can also run the following command to query the **libfuse.so**:
>
> ```
> find / -name libfuse.so*
> ```

- CentOS 7

  ```
  yum install -y openssl-devel fuse fuse-devel
  ```

- Ubuntu 16

  ```
  apt-get install -y libfuse-dev libcurl4-openssl-dev
  ```

Verification

Go to the directory where obsfs resides and run the following command to verify whether obsfs can run successfully:

```
./obsfs --version
```

If the obsfs version information is displayed in the command output, obsfs is running successfully. You can start **initializing** and **using obsfs**. **libfuse Installation Guide**

2.3 Initializing obsfs

Before using obsfs, you need to write the access key (AK and SK) information into the key file, enabling user authentication before using parallel file systems.

This topic describes how to write the access key information to the **/etc/passwd-obsfs** file.

Prerequisites

You have obtained the access keys (AK and SK). For details, see **Resource Preparation**.

Procedure

Step 1 Open the CLI.

Step 2 Run the following command to write the access keys into the **/etc/passwd-obsfs** file:

```
echo AK:SK > /etc/passwd-obsfs
```

Step 3 Run the following command to check whether the key information is successfully configured:

```
cat /etc/passwd-obsfs
```

Step 4 Run the following command to set the key file to be used only by the current user:

```
chmod 600 /etc/passwd-obsfs
```

----End

3 Operation Guide

3.1 Mounting a Parallel File System

Before using obsfs to manage your objects on OBS, you need to mount an OBS parallel file system to your local file system. Once the parallel file system is mounted, it works like a local directory. For example, you can upload an object to OBS simply by saving it to the mount point.

Constraints

- **obsfs only supports the mounting of OBS parallel file systems but not buckets.**
- The local directory to which the parallel file system is mounted needs not to be empty. But its original files will be unavailable after the parallel file system is mounted. They are recovered when the parallel file system is unmounted. Therefore, it is recommended that you use an empty directory as the target directory for mounting a parallel file system.
- Files or folders in the mounted directory do not support hard link commands.
- The number of directory levels cannot exceed 45.
- If a parallel file system is manually mounted by running a command, it needs to be mounted again every time the Linux is restarted.
- The actual size and used space of a parallel file system vary with your actual usage.

  NOTE

  You can configure auto mounting of a parallel file system upon startup. For details, see **Mounting a Parallel File System Automatically upon Startup**.
- A parallel file system can be mounted to multiple ECSs.

Prerequisites

- obsfs has been obtained. For details, see **Downloading and Installing obsfs**.
- obsfs has been initialized. For details, see **Initializing obsfs**.
- The mount point already exists in the local file system.

NOTE

You can run the **mkdir** command to create a directory for mounting a parallel file system. For example, to create a directory named **mountpoint**, the command is:

```
mkdir mountpoint
```

Mounting a Parallel File System Manually

Step 1 Open the CLI.

Step 2 Run the **cd** command to go to the directory where obsfs is located.

Step 3 Run the mount command in the following format:

```
./obsfs Parallel file system name Local mount point -o url=Region endpoint address -o passwd_file=Key file path -o use_ino Other mount parameters
```

- Parallel file system name: name of the parallel file system to be mounted
- Mount point: the absolute or relative path of the directory on which a parallel file system is to be mounted
- Region endpoint address: **Obtain the region information and endpoint address according to the region where the mounted parallel file system resides.**
- Mount parameters:

  Mount parameters, excepting **passwd_file** and **use_ino**, are optional. Select proper parameters according to your actual needs. The format is as follows: **-o *Parameter*=*Value***. If you do not need to specify a parameter value, use only **-o *Parameter***. Table 3-1 lists OBS mount parameters and their descriptions.

  NOTE

  For details about obsfs mount parameter configurations, run the **./obsfs --help** command to learn more.

Table 3-1 obsfs mount parameters

Parameter	Value	Description
passwd_file	Type: String Value: Set this parameter to the path of the access key file configured in **Initializing obsfs**.	The access key file contains the AK and SK information required for access obsfs.
use_ino	Required	If this parameter is specified, obsfs allocates the **inode** number.
big_writes	Not required	After the configuration, the maximum size of the cache can be modified.

Parameter	Value	Description
max_write	Type: Integer Recommended value: **128 KB**	This parameter is valid only when **big_writes** is configured. The default write size is 4 KB. The recommended value is 128 KB.
nonempty	Not required	This parameter allows the mount point to be a non-empty directory.
allow_other	Not required	This parameter allows other users to access the parallel file system.
max_background	Type: Integer Recommended value: **100**	You can use this parameter to set the maximum number of waiting requests in the background.
use_path_request_style	Not required	Indicates the path-based access (legacy API). If this parameter is configured, you need to use the specified endpoint address to access OBS. **NOTE** If the value of the URL in the command is an IP address, this parameter must be carried in the mount command. Otherwise, a domain name resolution error is reported.
umask	Not required	This parameter masks the file permissions. The **umask** command is executed upon each access to the file system, and automatically configures the mask for new permissions.

Parameter	Value	Description

obsfslog	Not required	This parameter enables the loading of log configuration parameters from the configuration file, including the log mode and level. You can change the log level without restarting the process when this parameter is enabled.
		NOTE
		The path of the configuration file is: **/etc/obsfsconfig**. If you want to use this parameter, you need to manually create the configuration file path on the local host and write the content by referring to the following example and comments.
		An example of the configuration file content:
		//Log mode. The value **0** indicates that the log mode is displayed on the client. If the value is **1**, the log file is generated in the following path: **/var/log/obsfs**, which is named after the process ID. If the value is **2**, the log is written into the system log.
		dbglogmode=1
		//Log level. **dbg**, **info**, **warn**, **err**, **crit**.
		dbglevel=info

Step 4 Run the following command to view the result:

```
df -h
```

If the information similar to the following is displayed, the parallel file system is successfully mounted:

```
Filesystem      Size      Used      Avail      Use%      Mounted on
obsfs           256T      0         256T       0%        /path/to/mountpoint
```

NOTE

The actual size and used space of a parallel file system vary with your actual usage.

If the preceding information is not displayed, the parallel file system fails to be mounted. In this case, you can add the following parameters to the command, so that the mount process and detailed debug logs are displayed in the command output:

```
-d -d -f -o f2 -o curldbg
```

Then you can locate the fault based on the error information. For details about troubleshooting of common faults, see **Failure in Mounting a Parallel File System**.

----End

Command Example

In this example, the **filesystem001** parallel file system and the access key file path **/etc/passwd-obsfs** are used as the prerequisites for mounting an OBS parallel file system.

- Mount the parallel file system **filesystem001** to the mount point **/mnt/mount_path/** and allow the directory to be non-empty.
  ```
  ./obsfs filesystem001 /mnt/mount_path/ -o url=obs.ru-moscow-1.hc.sbercloud.ru -o passwd_file=/etc/passwd-obsfs -o big_writes -o max_write=131072 -o nonempty -o use_ino
  ```
- Mount the parallel file system **filesystem001** to the **/mnt/mount_path/** directory and set the write cache size.
  ```
  ./obsfs filesystem001 /mnt/mount_path/ -o url=obs.ru-moscow-1.hc.sbercloud.ru -o passwd_file=/etc/passwd-obsfs -o big_writes -o max_write=131072 -o max_background=100 -o use_ino
  ```
- Mount the **filesystem001** parallel file system to the mount point **/mnt/mount_path/** and allow other users to access the directory.
  ```
  ./obsfs filesystem001 /mnt/mount_path/ -o url=obs.ru-moscow-1.hc.sbercloud.ru -o passwd_file=/etc/passwd-obsfs -o big_writes -o max_write=131072 -o max_background=100 -o use_ino
  ```

NOTE

If the value of this parameter is an IP address, add the **-o use_path_request_style** parameter to the mount command. Otherwise, a domain name resolution error is reported.

Mounting a Parallel File System Automatically upon Startup

The following uses CentOS 7 as an example to describe how to configure automatic mounting upon system startup.

Step 1 Log in to the Linux OS as user **root**.

Step 2 Run the following command to create a **/home/startobsfs.sh** script:

```
vi /home/startobsfs.sh
```

Step 3 Press **i** to enter the editing mode, and write the following comments and mount commands to the **/home/startobsfs.sh** script.

```
#!/bin/bash
cd obsfsabsolute path
./obsfs Parallel file system name Local mount point -o url=Region endpoint address -o passwd_file=Key file path -o use_ino Mount parameters
```

NOTE

For details about the mount command parameters, see **Step 3** in section "Mounting a Parallel File System Manually."

Step 4 Press **Esc** and enter **:wq** to save and exit the script.

Step 5 Run the following command to grant the **root** user the permission to execute the script:

```
chmod +x /home/startobsfs.sh
```

Step 6 Run the following command to grant the **root** user the permission to execute the **rc.local** file:

```
chmod +x /etc/rc.d/rc.local
```

NOTE

The location of the **rc.local** file varies depending on the OS. In CentOS 7, the file is stored in **/etc/rc.d/rc.local**.

Step 7 Add the following command to the end of the **/etc/rc.d/rc.local** file, save the file, and exit:

```
bash /home/startobsfs.sh
```

NOTE

If the **/etc/rc.d/rc.local** file contains an exit command, for example, **exit 0**, add the command prior to the exit command.

Step 8 Restart and verify.

After the Linux OS is restarted, run the following command to check whether the parallel file system is successfully mounted:

```
df -h
```

NOTE

If you want to cancel the automatic mounting of the parallel file system upon startup, delete the script of the obsfs mounting command in the **rc.local** file.

----End

Follow-up Procedure

Before adding, modifying, or deleting a parameter setting for a mounted parallel file system, you need to unmount the parallel file system and then mount it again. For details about how to unmount a parallel file system, see **Unmounting a Parallel File System**.

3.2 Unmounting a Parallel File System

If you do not need to access a mounted parallel file system or you need to add, modify, or delete parameters of the mounted parallel file system, you can unmount the parallel file system.

NOTE

Unmounting a parallel file system does not affect data stored in the parallel file system.

Prerequisites

The to-be-unmounted parallel file system is in the mounted state.

Procedure

Step 1 Open the CLI.

Step 2 Check whether the current directory is the mount point.

- If yes, exit the mount point. Then go to **Step 3**.
- If not, go to **Step 3**.

Step 3 Run the following command to unmount a parallel file system:

```
umount Local mount point
```

Step 4 Run the following command to check whether the bucket has been unmounted:

```
df -h
```

----End

4 FAQs

Can I Mount a Parallel File System to Multiple Clients?

Yes.

Can Multiple Parallel File Systems Be Mounted to the Same Client?

Yes. Each mounting process is independent.

What Can I Do If Error Message "error: conflicting types for 'int64_t'" Is Displayed When Compiling libfuse.2.9.7 on an Arm Platform?

Modify the content in lines 92 and 93 of the **include/fuse_kernel.h** to the following:

```
typedef unsigned long long __u64;
typedef long long __s64;
```

Figure 4-1 and Figure 4-2 are examples of the files before and after the modification.

Figure 4-1 File content before the modification

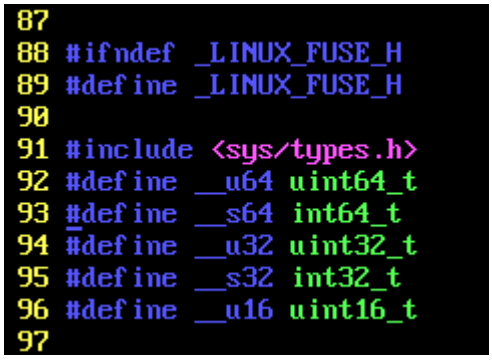
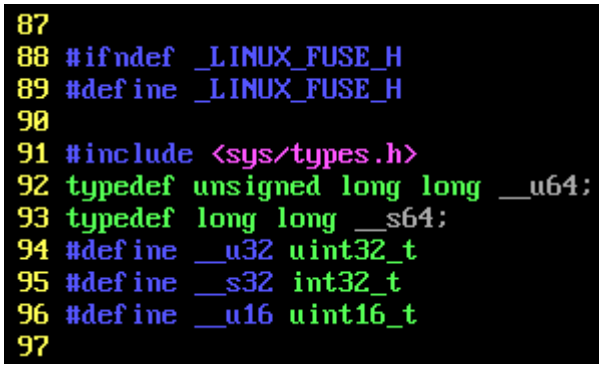


Figure 4-2 File content after the modification



How Can Other Users Access a Mounted Parallel File System?

Generally, only the user who executes the mounting has the access permission to the mount directory.

To authorize other users the access permission to the mount directory, you can specify the **allow_other** parameter in the mount command. The following is an example:

```
./obsfs filesystem001 /mnt/mount_path/ -o url=obs.ru-moscow-1.hc.sbercloud.ru  -o passwd_file=/etc/passwd-obsfs -o big_writes -o max_write=131072 -o allow_other -o use_ino
```

5 Troubleshooting

5.1 Failure in Mounting a Parallel File System

This topic describes the possible causes of and solutions to common failures in mounting parallel file systems.

Symptom 1: "The specified bucket does not exist" or Similar Errors Occur When Mounting a Parallel File System

Fault locating:

The parallel file system to be mounted does not exist.

Solution:

Create a parallel file system and mount it again. For details about how to create a parallel file system, see **Creating a Parallel File System**.

Symptom 2: "file system not support this request: this bucket not support filesystem" or Similar Errors Occur When Mounting a Parallel File System

Fault locating:

The mounted bucket is not a parallel file system.

Solution:

Create a parallel file system and mount it again. For details about how to create a parallel file system, see **Creating a Parallel File System**.

Symptom 3: "The request signature we calculated does not match the signature you provided. Check your key and signing method." or Similar Errors Occur When Mounting a Parallel File System

Fault locating:

Authentication failed. The access keys (AK and SK) are incorrect.

Solution:

Check whether the AK and SK information in the key file is correctly configured. For details about the check method, see **Step 3** in the section "Initializing obsfs."

Symptom 4: "specified passwd_file is not readable" or Similar Errors Occur When Mounting a Parallel File System

Fault locating:

The key file path configured in the **passwd_file** file is incorrect.

Solution:

Check whether the key file path configured in the **passwd_file** file is correct. For details about the check method, see **Step 3** in the section "Initializing obsfs."

Symptom 5: "Access Denied" or Similar Errors Occur When Mounting a Parallel File System

Fault locating:

Authentication failed. The IAM user does not have OBS operation permissions.

Solution:

Authorize the user with required OBS operation permissions. For details, see **Resource Preparation**.

Symptom 6: "unable to access MOUNTPOINT /obsfs: Transport endpoint is not connected" or Similar Errors Occur When Mounting a Parallel File System

Fault locating:

The mount point has been occupied by another obsfs mount process.

Solution:

Run the **df** command to query the partitions and points that have been mounted to. Then select an unoccupied mount point and mount the bucket again.

Symptom 7: "unknown option" or Similar Errors Occur When Mounting a Parallel File System

Fault locating:

One or more parameters in the mount command are incorrectly set and therefore failed to be identified by obsfs.

Solution:

Check the mount parameters by referring to **Table 3-1** in the section "Mounting the Parallel File System".

Symptom 8: "unable to access MOUNTPOINT /mnt/obsfs: No such file or directory" or Similar Errors Occur When Mounting a Parallel File System

Fault locating:

The mount directory does not exist.

Solution:

Use the **mkdir** command to create a directory or mount the bucket to an existing directory.

5.2 Failure in Unmounting a Parallel File System

This topic describes the possible causes of and solutions to common failures in unmounting parallel file systems.

Symptom 1: "target is busy" or Similar Errors Occur When Unmounting a Parallel File System

Fault locating:

During unmounting, the CLI terminal stays in the mount directory or its subdirectory.

Solution:

Exit the mount directory.

6 Related Operations

6.1 libfuse Installation Guide

This section uses libfuse 2.9.7 as an example target version to describe how to update libfuse on CentOS 7, SUSE 12, and Ubuntu 16.

Prerequisites

The source code of libfuse has been downloaded and then compiled to generate libfuse.so.2.9.7.

Download address of libfuse: **https://github.com/libfuse/libfuse/releases/tag/fuse-2.9.7**

```
tar -zxvf fuse-2.9.7.tar.gz
cd fuse-2.9.7/ && ./configure && make && make install
echo -e '\n/usr/local/lib' >> /etc/ld.so.conf
ldconfig
```

Installing libfuse on CentOS 7

libfuse.so.2.9.2 is installed on CentOS 7 by default.

Ensure that you have downloaded the libfuse source code and generated libfuse.so.2.9.7 after compilation as described in **Prerequisites**. Then perform the following steps.

Step 1 Run the following command to locate the links of the libfuse.so.2.9.2 library:

```
find / -name libfuse.so*
```

Step 2 Run the following command to copy libfuse.so.2.9.7 to the directory where the libfuse.so.2.9.2 library resides.

```
cp /usr/local/lib/libfuse.so.2.9.7 /usr/lib64/
```

Step 3 Run the following command to remove all links of the libfuse.so library of the earlier version:

```
rm -f /usr/lib64/libfuse.so
rm -f /usr/lib64/libfuse.so.2
```

Step 4 Run the following command to create links of libfuse.so.2.9.7 library similar to the deleted ones of the libfuse.so library of the earlier version:

```
ln -s /usr/lib64/libfuse.so.2.9.7 /usr/lib64/libfuse.so
ln -s /usr/lib64/libfuse.so.2.9.7 /usr/lib64/libfuse.so.2
```

----End

Installing libfuse on SUSE 12

libfuse.so.2.9.3 is installed on SUSE 12 by default.

Ensure that you have downloaded the libfuse source code and generated libfuse.so.2.9.7 after compilation as described in **Prerequisites**. Then perform the following steps.

Step 1 Run the following command to locate the links of the libfuse.so.2.9.3 library:

```
find / -name libfuse.so*
```

Step 2 Run the following command to copy libfuse.so.2.9.7 to the directory where the libfuse.so.2.9.3 library resides.

```
cp /usr/local/lib/libfuse.so.2.9.7 /usr/lib64/
```

Step 3 Run the following command to remove all links of the libfuse.so library of the earlier version:

```
rm -f /usr/lib64/libfuse.so.2
rm -f /lib64/libfuse.so.2.9.3
```

Step 4 Run the following command to create links of libfuse.so.2.9.7 library similar to the deleted ones of the libfuse.so library of the earlier version:

```
ln -s /usr/lib64/libfuse.so.2.9.7 /usr/lib64/libfuse.so.2
ln -s /usr/lib64/libfuse.so.2.9.7 /lib64/libfuse.so.2.9.7
```

----End

Installing libfuse on Ubuntu 16

libfuse.so.2.9.4 is installed on Ubuntu 16 by default.

Ensure that you have downloaded the libfuse source code and generated libfuse.so.2.9.7 after compilation as described in **Prerequisites**. Then perform the following steps.

Step 1 Run the following command to locate the links of the libfuse.so.2.9.4 library:

```
find / -name libfuse.so*
```

Step 2 Run the following command to copy libfuse.so.2.9.7 to the directory where the libfuse.so.2.9.4 library resides.

```
cp /usr/local/lib/libfuse.so.2.9.7 /lib/x86_64-linux-gnu/
```

Step 3 Run the following command to remove all links of the libfuse.so library of the earlier version:

```
rm -f /lib/x86_64-linux-gnu/libfuse.so.2
```

Step 4 Run the following command to create links of libfuse.so.2.9.7 library similar to the deleted ones of the libfuse.so library of the earlier version:

```
ln -s /lib/x86_64-linux-gnu/libfuse.so.2.9.7 /lib/x86_64-linux-gnu/libfuse.so.2
```

----End

A Change History

Release Date	What's New
2021-06-30	This is the first official release.